COMPUTE

Microprocessor Users Group Newsletter

Volume 4 Number 5 | November/December 1978 | Price $1.50

TABLE OF CONTENTS

SERIES 8000 HANDBOOK NOW AVAILABLE

A new Series 8000 Microprocessor Family Handbook is now available from National. The handbook replaces the N8080 System Design Manual.

The Series 8000 Microprocessor Family encompasses not only the INS8080A, but also many peripheral control components as well. The title change reflects this increased emphasis on peripherals.

The handbook provides design information pertaining to the National Semiconductor Series 8000 Microprocessor Family. The material contained in the handbook is presented at a level-of-detail sufficient for use in the design and development of microprocessor systems using the proven INS8080A CPU and support components.

A limited number of Series 8000 handbooks are available, priced at $5.00, and can be ordered from:

Marketing Services, MS/520
National Semiconductor Corp.
2900 Semiconductor Drive
Santa Clara, CA. 95051

Orders must be received by January 31, 1979.

NATIONAL DEVELOPS ENHANCED 8048 FAMILY OF SINGLE CHIP MICROCOMPUTERS USING ADVANCED XMOS PROCESSING

Using an advanced process called XMOS, National Semiconductor Corporation has developed an enhanced version of the industry-standard 8048 family of single-chip eight-bit microcomputers.

The family, which not only includes the industry-standard 8048, 8049, 8035 and 8039, but proprietary high-density versions called the 8040 and 8050 as well, is designed to be lower in power, smaller in chip size and higher in density than similar devices on the market.

Centerpiece of the new family is the proprietary INS8050, which contains 4K bytes of read only memory, 256 bytes of random access memory on the same chip as the CPU---twice the capacity of any similar single-chip microcomputer now on the market.

The INS8050 now makes it possible for the designer to upgrade his system features without making artwork changes to his system or having to relocate the software he has presently generated for his 8048/49 design. In today's marketplace the usage of more memory is especially relevant now as it translates itself to more product features and thus added value to the O.E.M.

Previous generations of the 8048 family available from other manufacturers use a semi-standard n-channel MOS processing which limits their speed, and increases their power consumption, ruling them out in many single-chip microcomputer applications.

In National's INS8048 family, these constraints are eliminated through the use of its powerful XMOS process. As a result, 8048 family devices which are 15 to 20% smaller in chip size, consume 20 to 25% less power in full operation and 12 to 35 times less power in the standby mode. Standby voltage is only 2.2 volts, less than half that required for present NMOS 8048 devices.

With this family of low-cost one-chip microcomputers, a product designer can now add an almost endless variety of sophisticated processing and control functions to his

Continued Page 2

Users Group Director ..... Georgia Marszalek
Applications Consultant ..... Keith Winter
Graphic Artist ..... Marva Moulton
Editor ..... Jim Teller
Phototypesetter ..... Patti Hudson
Sponsored by National Semiconductor Corporation, 2900 Semiconductor Drive, Santa Clara, Calif. 95051

product line. The microcomputers can be programmed to perform all the operations generally required in products such as home appliances, electronic games, automotive applications, small business machines, instruments, data terminals, vending machines and numerous other types of control functions.

Six Single-Chip Microcomputers

The new family from National will initially include six single-chip microcomputers, all fully programmable systems designed to perform input/output control and processing tasks at rates up to 400,000 operations a second:

- INS8048, which can operate as a standalone single-chip system with programs stored in 1 kilobyte (8K bits) of masked ROM on the chip and 64 bytes of RAM data memory;
- INS8049, with 2K bytes (16K bits) of masked ROM and 128 bytes of RAM on chip;
- INS8050, National's proprietary, high-density single-chip microcomputer with 4K bytes (32K bits) of masked ROM, pin compatible with the 8048/8049; and 256 bytes of RAM on chip;
- INS8038, INS8039, and INS8040, ROMless versions of the INS8048, INS8049, and INS8050.

All six devices contain the same basic processing and control systems in pin-compatible 40-pin packages---8-bit general-purpose central processor with internally expandable register and stack array, 96 instructions and a 2.5 microsecond cycle time for the 6 MHz part and 1.36 microseconds for the 11 MHz part; three programmable I/O ports and eight other control and timing lines; programmable interval timer/event counter; priority interrupt controls; system clock generator; and a full set of generally required systems controls and utilities.

The central processor (CPU) on each chip can operate as a byte processor (eight-bit parallel binary); 4-bit binary or BCD arithmetic processor. Program and data memories are independently addressable and expandable.

MICROBUS™ and System Expansion

All six microcomputers are expandable with National's family of peripherals and standard memory components via MICROBUS.

The MICROBUS concept developed by National provides a universal component-level system bus for communication among the company's various microprocessor and microcomputer products, memories and peripherals.

Application of the MICROBUS standard will provide the user with a broad line of support devices which can effectively be "plugged" into systems without regard to complex timing or electrical analysis.

System Support and Availability

National will support the INS8048/35, INS8049/39, and INS8050/40 single-chip microcomputers with the STARPLEX™ development system, a disk-based software development system which supports National's full line of microprocessors. The In System Emulator (ISE), a sophisticated development tool, works with the STARPLEX to check out software and perform final systems integration.

NIBASM - A NEW SC/MP ASSEMBLER

NIBASM is a two-pass SC/MP assembler written in NIBL (National Industrial BASIC Language) that may be run on either a SC/MP LCDS or homebrew system configured to run NIBL. NIBASM allows SC/MP users to assemble programs in a symbolic language that is a subset of the standard SC/MP assembler language. NIBASM accepts source input from the console keyboard or paper tape reader.

NIBASM requires 4K of RAM or ROM on any page 2 through 7; page 1 should contain 4K of RAM for source code test.

NIBASM recognizes all SC/MP opcode mnemonics and the .BYTE and .END directives. The Assignment Statement

symbol=expression

assigns the value of an eight-bit expression on the right of the equals sign to the symbol on the left of the equals sign. Assignment statements can also use the period (.) as a special symbol to specify the location counter. For example, .= sets the assembler's program counter to the value to the right of (.=), and .=. sets a label to the left of (.=.) to the assembler's current PC value. If a value is present to the right of (.=.) then the object code counter is incremented by that amount.

NIBASM was written by:

Roger A. Marin
Tektronix, Inc., M/S 58-594
P. O. Box 500
Beaverton, Oregon 97077

Program documentation and listing are available free by specifying Program Number SL0060 on the User's Library Order Form.

SYSTEM PACKAGE AIDS SERIES/80 SYSTEM ASSEMBLY

A new, completely assembled, rack-mounted system accepts up to eight Series/80 microcomputer, memory, and interface boards. Designated the RMC660, the unit consists of two system board chassis, power supply, and front panel with control switches mounted in a standard RETMA enclosure. It may be used either to assemble a Series/80 board-level computer system or as an expansion chassis on existing systems. Boards need only be inserted in the slots-power, bus, and control lines are interconnected with etched backplane circuits, allowing convenient expansion to meet customer needs.

EXPAND INS8060 MULTIPROCESSOR SYSTEMS USING PRIVATE/SHARED BUSSES

Keith Winter
NSC. Santa Clara. CA.

System throughput for microprocessor systems can be increased by implementing a multiprocessor configuration with the INS8060 microprocessor. This configuration can increase raw system throughput and/or the number of tasks performed. However, there is a practical limit to the total number of processors that can share a bus before bus saturation is reached (refer to Application Note AN-197 for analysis). This article shows a proposed solution to this problem.

CONCEPT OF PRIVATE/SHARED BUSSES

Theoretically, if each processor in a multiprocessor system had its own memory and peripherals which only it could access, each one would operate at full speed. However this would eliminate most or all of the advantages of a multiprocessor system, since the individual processors could not communicate with each other or could, at best, communicate inefficiently through some type of serial data link network. Therefore, the idea is both to give each processor its own segment of memory that is private to it alone, thus allowing it to execute at full speed, and also give it access to system memory (for parameter passing, etc.) and to system peripherals. In this manner, each processor acts independently while working within its own memory area. Accesses to system resources are limited to only those times when it needs mass storage or some peripheral device.

The major benefit of this method of system configuration is that it effectively makes each processor look like a single CPU executing on a single bus. The only time it looks like a part of a multiple CPU system is when it must go outside its own private environment to pass parameters to other CPUs or to access a peripheral device, such as a floppy disc. When accessing data on the system bus, the INS8060 makes a bus request in the same way that it would in a standard multiprocessing system. When working from the internal or private bus, it appears to be alone and is granted bus access instantly.

SYSTEM IMPLEMENTATION

A suggested system implementation for one CPU module is shown in figure 1. Note that some of the internal bus-request/bus-grant circuitry is duplicated off-chip in discrete logic. A discussion of the sequence for gaining access to the busses follows.

When the INS8060 makes a bus request, whether it is for the private or the shared bus, it is granted bus access immediately. Bus grant holds true until after the desired address is valid. At this time, if it is determined that the access is to be internal, no action is taken by external logic and the INS8060 continues with the current cycle. If, however, it is determined that the access is to be to the external or shared bus, the NENIN signal is taken away from the INS8060, thus denying bus grant, until it is determined that the external bus is free. When the external bus is free for an I/O transfer to take place, NENIN is again driven true, and the INS8060 continues its current bus cycle. This is possible due to the internal architecture of the INS8060. If NENIN is removed before an I/O transfer takes place, the INS8060 will wait for NENIN to become true again and then restart the operation.

A simple address select performs the function of determining if the access is internal or external. The system designer would choose a point in memory which is the cutoff between private and shared busses and would set the address select accordingly. For example, he might choose a point at address X'800. This division of memory is shown in the system of figure 1. In this system, all private memory would be from X'800 to X'FFF. This assumes the designer is not latching the upper four address bits from the data bus at status time and is working within a single page. The designer could, however, latch the upper four address bits and select the center of the INS8060's address space (X'8000) as the cutoff, thereby allowing varying amounts of private memory (up to 32K) per CPU module and have all system resources above this point. In this way he could tailor each module to some dedicated job requiring more or less memory but not change any of the private/shared bus architecture.

By this technique, it can be seen that many more than the usual limit of four or five processors can be put on the system bus. Since it is likely that each processor will be making only infrequent access to the shared bus, that bus should not become saturated until a great many processors are tied into the system.

ADDITIONAL POSSIBILITIES

Once the system has been designed around this concept, further possibilities arise. For example, with the addition of bidirectional buffers on the address bus as well as on the data bus, a master CPU module could down-load programs into other CPU modules defining different jobs for each one to perform along with the data needed to execute that job. A block diagram of a system of this type is shown in figure 2. In order for this system to work properly, the master CPU must be able to stop any individual slave's execution while the down-loading operation is accomplished. This could simply be a signal that allows the master to selectively initialize the slaves. In this manner, the slave's busses are tri-stated and the master can down-load unhindered by the slave's operation. A method of selecting a slave's buffers and enabling a write operation would also need to be provided.

See diagrams on pages 4 and 5

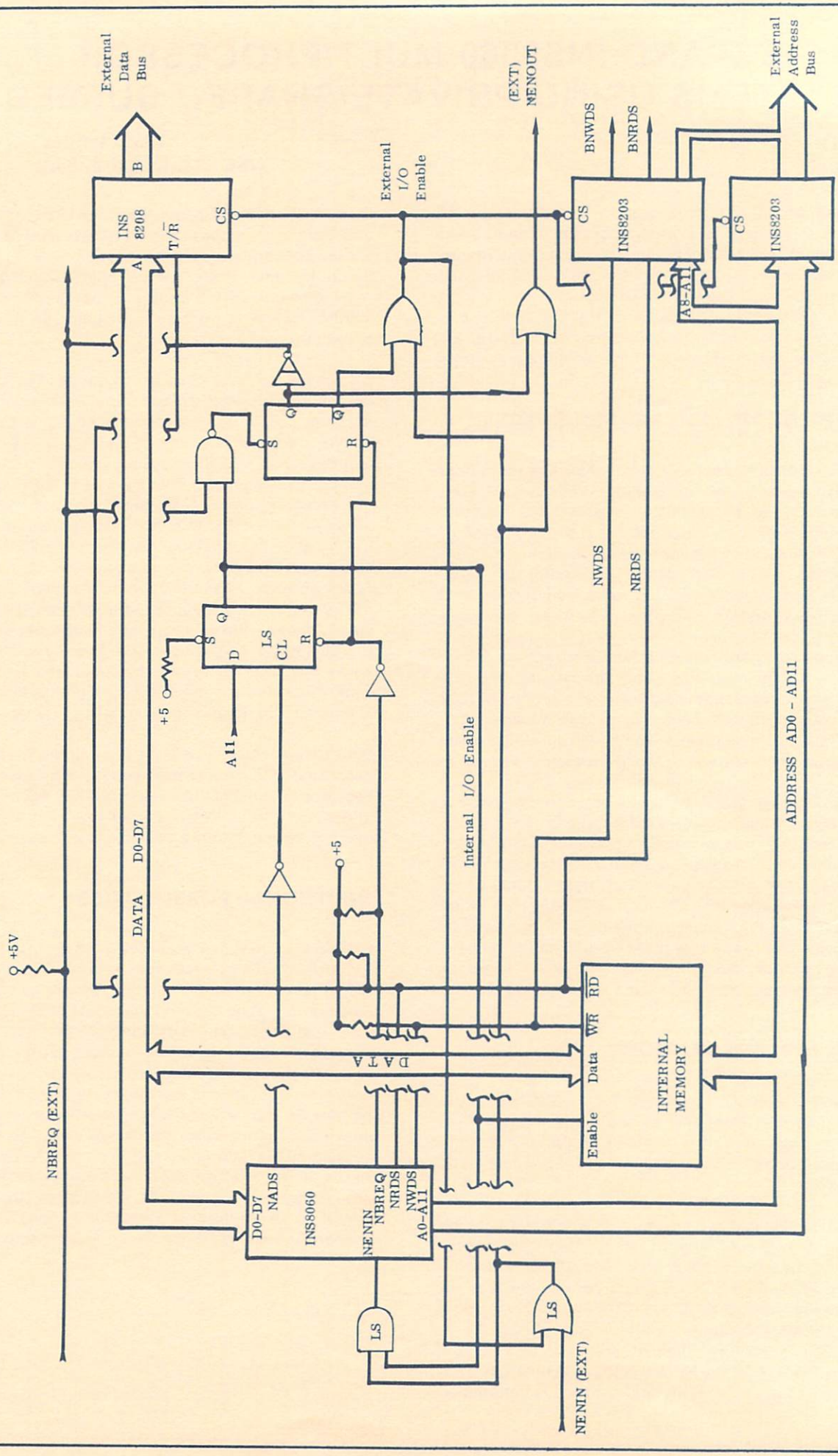


Figure 1. Private/Shared Bus Implementation

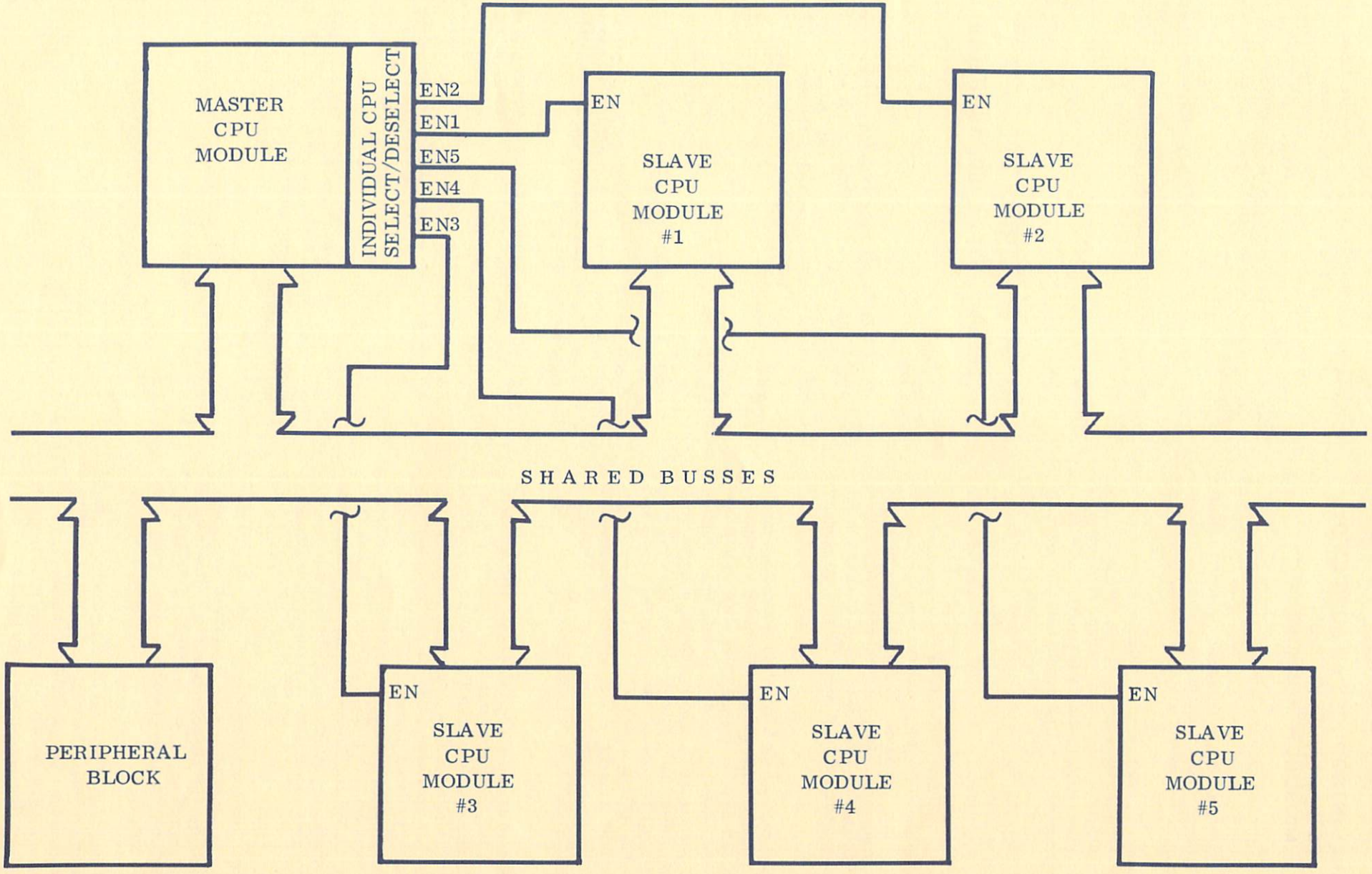


Figure 2. Master/Slave With Down Load Capability

NATIONAL INTRODUCES AN ADVANCED COP400 SERIES OF 4-BIT MICROCONTROLLERS

National is now producing a wide-ranging and sophisticated family of 4-bit single-chip microcomputers aimed at meeting the total needs of the low-end controller marketplace.

Designated the COP400 series of single-chip microcontrollers, the family initially includes a dozen members, each of which is a complete "computer-on-a-chip," containing all system timing, internal logic, ROM, RAM, and I/O necessary to implement dedicated control functions in a wide variety of applications. The COP400 family of devices feature instruction sets, internal architectures, and I/O schemes designed to ease keyboard input/display output and efficient BCD data manipulation. On-chip RAM sizes range from 512 x 8 to 2,048 x 8 bits, RAM from 32 x 4 to 128 x 8 bits, instruction sets from 43 to 57 commands, I/O lines from 16 to 36, and instruction cycle execution times ranging from 4 to 16 microseconds. The family is fabricated using three processes: an advanced, high-speed n-channel MOS, a low-power nMOS, and an even lower power complementary MOS. Prices begin at $.99 in volume for the smallest member of the family.

The various members of the COP400 family allow the user to specify an optimum microcontroller for use in a particular application. Not only can the user pick a part with RAM, ROM, I/O and speed optimized for specific tasks, but the family also offers a choice of parts with differing electrical characteristics. Each part contains a number of clock, I/O and other options, mask programmed into the device at the same time the ROM is coded with the user's dedicated program. This allows great flexibility in matching particular COP400 microcontrollers to the user specifications, eliminating the need in most cases for external interface logic.

All COP400 devices feature single supply operation and fast standardized test procedures that verify the internal logic and user program. The flexible I/O configuration of the COP400 microcontrollers allow them to interface and drive a wide range of devices using a minimal amount of external parts. Typical interfaced devices include: keyboards and displays (direct segment and direct digit drive), external data memories, printers, other COP devices, A/D and D/A converters, power control devices such as SCRs and TRIACs, mechanical actuators, general-purpose microprocessors, shift registers, and external ROM storage devices.

The COP400 devices are aimed at such high-volume applications as clocks, timers, laboratory instruments, radio controllers, appliance controllers, programmable sequencers, scales, cash registers, calculators, microcontroller computational elements, toys, games, and automotive computers.

To aid in the efficient and speedy programming of the COP400 microcontrollers, National has developed the COP Product Development System (PDS), built around a 16-bit microcomputer, 32K bytes of R/W memory and 12K bytes of PROM firmware. The disk-based system features an editor and assembler for handling source code entry, conversion to object code and maintaining documentation. An in-circuit emulator card attachment allows object code to be executed under the careful control of a COP Monitor debug utility. The PDS also features a circuit fixture for incoming inspection of COP400 devices. In single unit quantities, the COP PDS is $3995.

Future members of the COP400 family will include expanded software and hardware capabilities, alternative electrical specification devices, and smaller devices suitable for use in less demanding applications.

GETTING OUT OF NIBL TROUBLES

by Colin Morle
NSC, Bedford, UK

Should you reinitialize NIBL (National Industrial BASIC Language) after you have entered a program or if you accidentally type NEW, you will find that all is not lost because all that these actions do is to set addresses 1120 and 1121 to FF to tell NIBL to start all over again. You can get around this by typing:

```
@#1120 = #00
@#1121 = #0A
```

This resets addresses 1120 and 1121 to their original states.

This technique works great if you want to leave NIBL to check out or write some software in machine code on the LCDS (Low Cost Development System) and then return to NIBL by typing the above.

The advantage this method has over the method given in the January/February edition of COMPUTE (Vol. 4, No. 1) is that no additional software is required.

RANDOM DATA

Although we are attempting to publish COMPUTE bimonthly, circumstances occasionally force us to delay publication of a particular issue. We will make every effort, however, to publish six issues per yearly volume, as in the past.

The Microprocessor Service Center has the same address but a new phone number.

Microprocessor Service Center
National Semiconductor Corp.
675 Almanor Avenue
Sunnyvale, CA. 94086
(408) 737-6279
or
(408) 737-6270

LIST ROUTINE FOR SC/MP

by D. G. Webster
Head Wrightson Process Engineering Ltd.
P. O. Box 101, Stockton-on-Tees
Cleveland, England
TS17 6AZ

After recently developing hardware and software for a SC/MP-based data logger with extensive number crunching capability (via a bus sharing MM57109 Number Cruncher), we came to the all-important though somewhat tedious task of software documentation.

As all the programming had been done in machine code on a Low Cost Development System (LCDS) via the keyboard and/or teletype, the only documentation that had been carried out during program development was the flowcharts created at design time and scribbled listings converting these flowcharts into mnemonics and machine code.

To turn these rough, and in some cases, uncorrected notes into presentable listings for documentation was going to be a very tedious and obviously error-prone operation.

Since we had available punched tape copies of all working programs and a teletype, capable of typing neat listings if correctly directed, it seemed a worthwhile project to spend a few days producing a program to run on the LCDS which would turn these thousands of bytes of machine code into presentable mnemonic listings.

To this end, a program was developed within the following design constraints.

(1) The program should be considered as an enhancement of DEBUG and can therefore use any of the DEBUG teletype routines as well as the RAM space on the LCDS not used by the DEBUG firmware.

(2) The overall size of the program should be not more than 512 bytes so that it could be burnt into a single 5204Q PROM. (This could then fit into the CPU card PROM socket.)

(3) The program should be loadable at any address within the 64K addressing capabilities of SC/MP, without alteration to any part of the program — within the limitations, of course, that it does not overlap a page boundary.

(4) The format of the printout should be of A4-size listings with suitable spacing between blocks of print to enable the continuous teletype paper to cut up into A4-size sheets for filing.

General Description

The program developed, LIST, occupies a total of 512 bytes, the first 282 of which are program area and the remaining 230 bytes form the 'search table'.

This table consists of 52 entries, each composed of an opcode base, a string of ASCII characters, and a terminator. Each opcode base covers a range of individual opcodes, the commonality between which is the characters to be printed. The amount by which a particular opcode is greater than the base code is used to generate the pointer number and mode of addressing character, i.e., immediate or @ indexed. LIST searches this table until it finds an opcode similar to the opcode to be identified.

When this testing is successful it prints the characters following that base code. If a particular base code similarity test fails, then the program looks for the next base code entry in the table by moving down the table until the terminator preceding the next opcode base is identified.

LIST can reside in any 512-byte continuous memory area that does not cross a page boundary and starts with a lower two-digit hexadecimal address of 00, i.e., a start address range of X000 to XE00, where 'X' is any hexadecimal digit. Once LIST has been assigned to a memory location, then all that is required to produce a listing of a RAM-or ROM-resident program is to commence execution at the address of the first byte of LIST, usually with a DEBUG go! command on the teletype.

LIST will then announce itself by typing : 'ADDRESS?' at which point the hexadecimal four-digit start address of the program area to be listed should be entered at the teletype, followed by CR. At this point LIST will leave a one-inch space, then proceed to write the address, code, and mnemonics of all sequential instructions starting at the address provided by the user.

After typing 60 lines of program it will then leave a 1 2/3" space before continuing to type the next block of the listing. This spacing arrangement allows successive blocks of typed program to be cut up into accurate A4-size pages.

LIST can be stopped at any point in its operation by pressing INIT thus returning to DEBUG mode.

The only restriction on the use of LIST is that the start address be that of a legitimate opcode and not a data byte. Any data blocks encountered during a listing will be treated as opcodes and a corrupted printout will probably result. Any opcode that LIST cannot identify would have the address and hexadecimal value of that byte typed, but a space left where the mnemonic should appear.

An additional useful feature of LIST is that any PC-relative jump instruction encountered in the listing has its second byte (displacement) replaced by the jump destination lower-half address, in the printout following the mnemonics. This enables easy checking of program flow.

Firmware Prerequisites to the Use of LIST

(1) DEBUG firmware resides at 7800 : 7FFF

(2) DEBUG RAM resides at 7700 : 77FF

(3) LIST assumes that the value of the program counter at entry (and hence its own address) can be found at the DEBUG PC save location of 77F5 and 77F6.

(4) RAM for stack use is available at a top address of 77D5.

A LIST program listing is available from COMPUTE.

The Bit ● Bucket

$\propto \phi \lambda o \eta \leq \bullet \times \Omega \sum \angle + \dagger \cong$

Dear COMPUTE:

In my system, whose outputs are address and memory contents, I find the delay instruction useful in outputting characters in legible form. The desired pattern is ANDed with hex 80 and the result made the displacement. That is a fringe benefit of slow memory chips and using a 555 for the system clock.

I really enjoyed "Micro Edit" in COMPUTE (Vol. 4, No. 3). Please send information on the INS8295, the SC/MP NIBL ROM.

Thank you,

Bill Von
4041 Oriole Ln.
Oxnard, CA. 93030

$\Delta \Phi \Xi \Lambda \Theta \Psi \Upsilon \amalg \Omega \delta \gamma \xi \mu \pi$

Dear COMPUTE:

You may be interested to know of our range of LCDS compatible peripheral boards and also our 8060 based development system.

The IDES development system uses a 20-key keypad and 32 seven-segment displays to provide input/output facilities and features single key MOVE and COPY capabilities, software for EPROM programming and fully transparent LCDS type monitor firmware. At under £100.00 (≃ $200) we feel that its capabilities would be invaluable to any group working with the 8060.

Our range of peripheral cards include a video interface, 4K 2708 EPROM card, cassette interface and many others and are designed to be plug compatible with LCDS and IDES backplanes where possible.

We have found the 8060 processor to be ideally suited for a large number of tasks and feel that the range of cards available now make prototyping and small scale production even easier.

Yours faithfully,

Dave Graham
Technical Manager
The Studio
Quarry Hill
Box, Corsham,
Wiltshire SN14 9HT

Dear Georgia:

I was pleased to see our 'Guide to SC/MP Programming' recommended by one of your correspondents in the May/June edition of 'Compute'. Your readers may find it helpful to know that we have moved to the address below and that the price is £3.50 for UK readers and £4.00 elsewhere; prices include postage.

We have just launched a SC/MP protoboard featuring your excellent RAM I/O chip. This board is the answer if you want a cheap, versatile micro-controller - 23 I/O lines and space for customising.

Yours sincerely,

Jon Drury
Managing Director
Kemitron Electronics
The White House
Warrington Rd., Chester CH2 3PA

Please contact Kemitron for information on their protoboard.

$6 \# \geq \div \ll \int \uparrow \leftrightarrow \Phi \Xi \Psi \Theta \Omega \infty$

Dear COMPUTE:

I am sure your readers will enjoy using the INS8298 LLL BASIC as much as I have during the last 6 months or so. However, their enjoyment will begin much sooner if they ground pin 4 of the 74LS138 address decoder rather than returning it to P1-26, the BLC (SBC) INH2/ signal, as shown in the article on page 24 of COMPUTE, Vol. 4, No. 4. This signal is not implemented on either National BLC or Intel SBC 80/10's and, when high or open, will render all attempts to access the MAXI-ROM unsuccessful. I presume that the low-speed paper tape punch command in the product literature has also been corrected.

Some of the multiprocessor cards like the 80/20, etc., may work as shown but I've had no direct experience with them.

An excellent example of the capabilities of the INS8298 appeared in the REVIEW OF SCIENTIFIC INSTRUMENTS recently - the application, if I remember correctly, involved the implementation of full three-mode temperature control.

Very truly yours,

Willaim A. Resch III
Users Group Electronics
Box 3312, Kingsport, Tenn. 37664

Bit Bucket continued . . . . . .

Dear COMPUTE:

I recently took a closer look at the program "Unsigned Multiply" of the "SC/MP Math Routines" (SL0027A). You had a comment on this program in Vol. 3, No. 4, page 10, saying: "...a CCL instruction should be inserted between lines 105 and 106." However, that is not enough to make the program work properly. You also have to remove the CCL instruction at line 108 and put it between lines 120 and 121. If not you will lose a possible carry from the addition at line 106. With this CCL instruction in the former position you will get that X'FF times X'FF is X'0001, while the correct answer should be X'FE01.

Sincerely,

Jon Kleiser
Kastellveien 9
N-Oslo 11, NORWAY

BROCHURE ON PERSONAL COMPUTER COMPONENTS AVAILABLE FROM NATIONAL

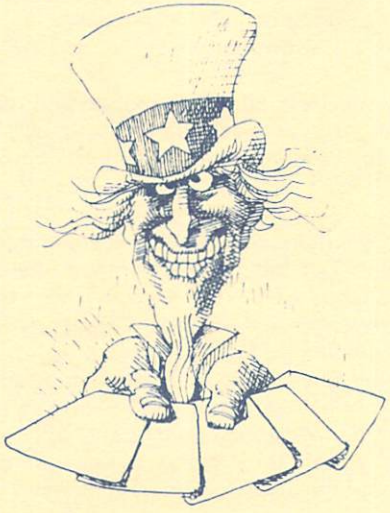
National Semiconductor Corporation, committed to providing a complete line of semiconductor products for personal computers, has a 24-page brochure available on its broad range of components to serve this industry.

The glossy-paged booklet is a guide to more than 100 components including microprocessors, memories, CRT Controllers, LED displays, floppy disk interfaces, serial and parallel interfaces, sound synthesizers, analog interfaces, and printer interfaces.

The brochure also discusses National's support for personal computer manufacturers.

To receive a free copy of this brochure, contact:

Wade Miracle MS/770
National Semiconductor Corp.
2900 Semiconductor Drive
Santa Clara, CA. 95051
(408) 737-5897



PACE LCDS CONVERSION FOR 8900

Dave Gibbs and Subhash Bal
National Semiconductor Corp.
Santa Clara, Calif.

This article describes a method of converting a PACE LCDS into an INS8900. All required changes are on the motherboard. The component side is shown in figure 1; the underside is shown in figure 2. The following symbols are used in the figures and supporting text:

Symbol	Meaning
TC	Trace Cut
NH	New Hole
SR	Solder Remove
SI	Solder Insert

Hardware Conversion

The necessary changes affect the following three items:

		PACE LCDS	8900
1.	Crystal:	2.667 MHz	4 MHz
2.	Power Supply and V_{BB}:	V_{GG} = -12V	V_{DD} = +12V
		V_{BB} = +8V	V_{BB} = -8V and GND supply
3.	Clock:	+5 to -12V	0 to +12V

Changes are accomplished as follows:

1. Crystal

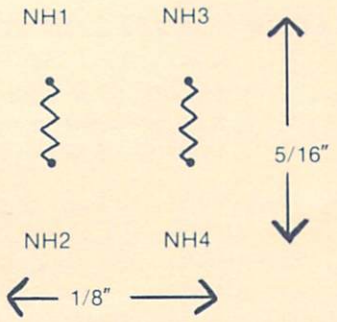
To remove the 2.667 MHz crystal from the PACE LCDS, remove solder from SR1 and SR2 (figure 2), and pull out the crystal. Now insert a 4MHz crystal (standing up on component side), and solder SI1 and SI2 (figure 2).

2. Power Supply and V_{BB}

Cut the old power and clock traces to accommodate new power lines. The traces to be cut on the component side are TC1 and TC2 (figure 1). The traces to be cut on the underside of the board are TC3, TC4, TC5, TC6, TC7, and TC8 (figure 2).

Install a jumper between TC7 and TC8. This jumper must not touch V_{DD} (figure 2).

Insert a voltage divider to produce the new V_{BB} (-8V). Use a number 60 drill bit to make four holes at NH1, NH2, NH3, and NH4 (figure 2). Make sure that holes are as far on the board as possible and spaced as follows:



The resistors must not short to the chassis of the motherboard. Insert a 2KΩ resistor between NH1 and NH2, and a 2.7KΩ between NH3 and NH4. Jumper NH2 to NH4 and also to V_{BB} (figure 2). Jumper NH1 to -12V (figure 2) and NH3 to GND (figure 2). This completes the hookup of the new V_{BB}.

Now solder in a .1μ F filter cap between +12V and GND (figure 2).

To connect the new V_{DD} simply solder in a jumper from +12V to V_{DD} (figure 2).

To connect the new V_{CC} simply solder in a jumper from +5V to V_{CC} (figure 2).

To connect the new V_{SS} connect GND to V_{SS} (figure 2).

This completes the 8900's power needs and leaves only the clock driver circuit.

3. Clock

Mount an eight-pin low-profile socket to the motherboard. Use a number 60 drill bit to make holes at NH5, NH6, NH7, NH8, NH9, NH10, NH11, and NH12 for the socket. Take care that none of the holes touch any existing traces and have the socket as far in on the motherboard as possible.

Insert the socket and solder a jumper from pin 8 (NH12, figure 2) to +12V. Make another jumper from pin 4 (NH8) to GND.

To hook up the clock input, insert a jumper from pin 1 (NH5, figure 2) to BCLKT*. To hook up the clock output, insert a jumper from pin 3 (NH7, figure 2) to CLKX. Now solder a 50pF cap between NH6 and NH7.

Insert a DS3642N into the socket with pin 1 at NH5.

Insert an IPC-16A/600D into the old PACE socket with pin 20 at V_{SS} (figure 2). This completes the hardware conversion.

Optional Firmware Changes

Firmware (represented by two ROMs on the board) must be updated to accommodate the higher clock speed for the 8900 only if teletype interface is used with the LCDS. If only the keyboard is to be used, no firmware change is required. The firmware modification is accomplished by the following alterations on four locations.

Address	Value	
F 239	1F7	110 baud
F 23A	171	150 baud
F 23B	B7	300 baud
F 23C	2B	1200 baud

To achieve this, the two firmware ROMs must be replaced by four PROMs (5204) in the PROM sockets, and the PROMs on the LCDS must be enabled. The program in PROM will be identical to the replaced ROMs except for the above four locations.

Figures 1 and 2 (PACE LCDS motherboard, component side and solder side) are shown on pages 11 and 12, respectively.

ARTICLES ABOUT SC/MP

A series of five articles published in Elektor magazine may be of interest to some SC/MP users. These articles, entitled "Experimenting with the SC/MP," appeared in the Elektor issues of November 1977 through March 1978 (Vol. 3, No. 11 through Vol. 4, No. 3). The topics discussed include programming techniques, SC/MP registers and address decoders, the memory extension card and interrupt system, the hexadecimal input/output unit, and "Elbug," the monitor software for the Elektor SC/MP system.

The address of Elektor is: Elektor Publishers Ltd., Elektor House, 10 Longport Street, Canterbury CT1 1PE, Kent, England (Telephone 0227 54430).

NATIONAL TO CONDUCT SEMINARS ON INTERFACE CIRCUITS IN 30 CITIES

From now until March 1979, National Semiconductor will conduct a seminar in 30 major locations on new interface designs and applications. Special emphasis will be on line driver/receiver fundamentals, safe operating range for peripheral drivers, and a new LSI circuit for CRT controllers.

The seminar will be a concentrated session, packed with circuit descriptions and applications, but will also allow opportunity for persons attending to ask about specific applications.

The seminar, geared to design engineers and technicians, will be presented in Baltimore; Boston; Buffalo; Chicago; Cleveland; Dallas; Denver; Detroit; Ft. Lauderdale; Indianapolis; Los Angeles; Minneapolis; Mountain View, CA; Orlando; Philadelphia; Phoenix; Pinebrook, NJ; Pittsburgh; Portland; Salt Lake City; San Diego; The San Fernando Valley; Santa Ana; Seattle; Spokane; Stamford; Toronto; Tuscon; Utica; and Westbury, Long Island. Interested parties should contact their local National Semiconductor sales office for exact time and locations.

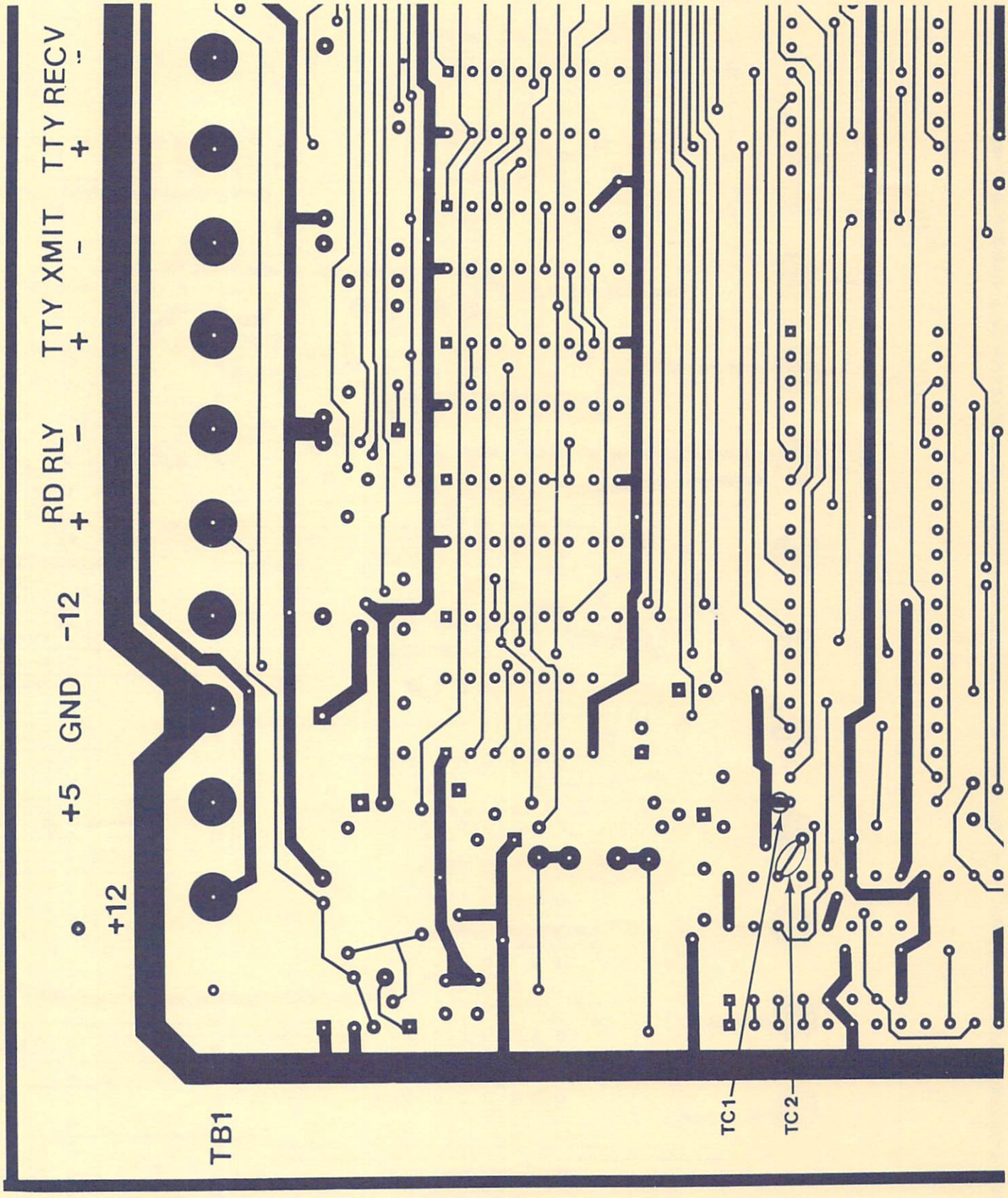


Figure 1. Layer 1 (Component Side)

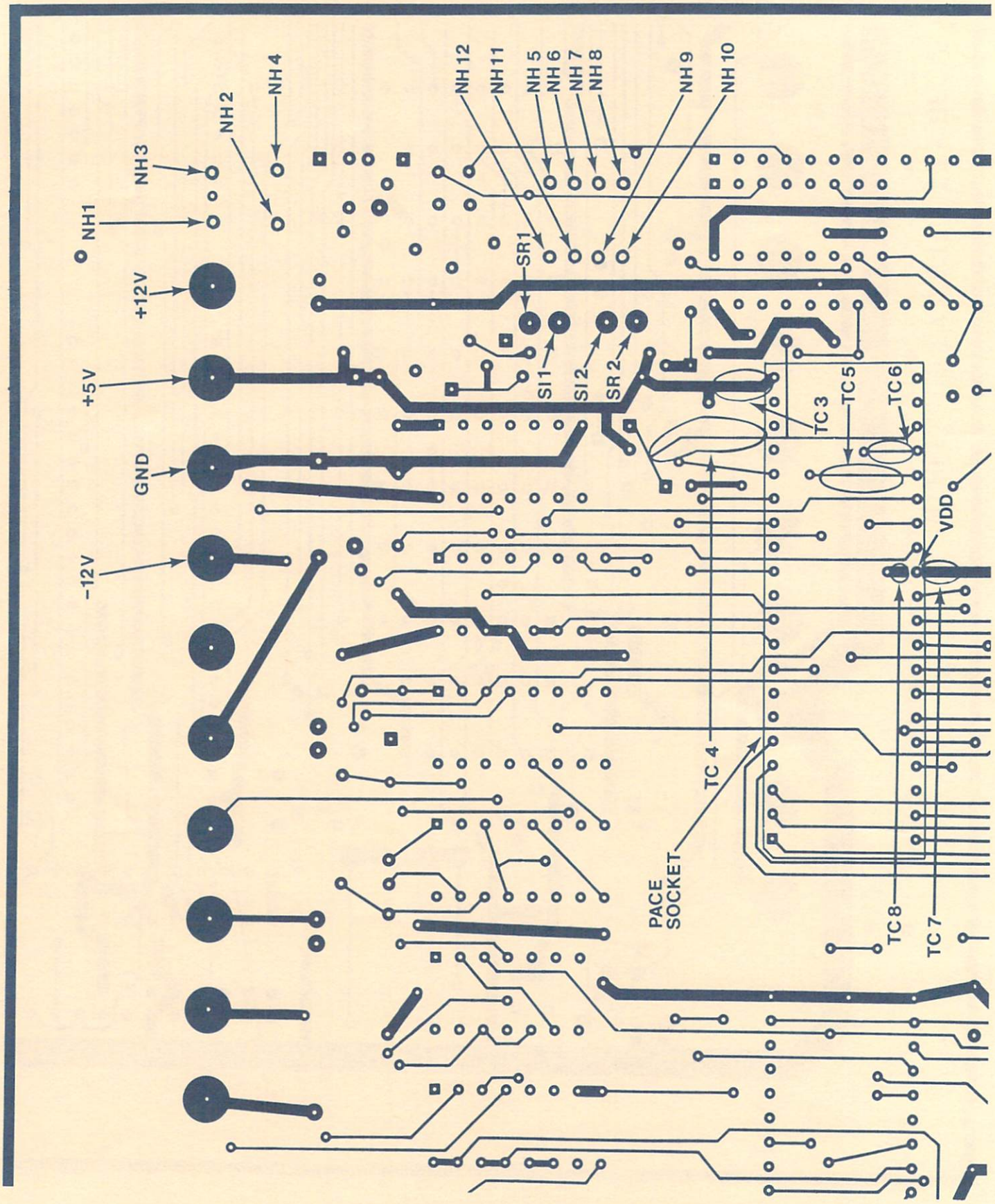


Figure 2. Layer 2 (Solder Side)

- NEW TRAINING COURSES - DESCRIPTIONS AND SCHEDULES

The birth of the STARPLEX™ has occurred at a very opportune time. It is also time for a new course schedule. We have designed three new courses around our new development systems. The Microcourse and STARPLEX™ User's Course utilize the STARPLEX™ development system; for the very low end, we have a new COPS course and development system. All of these courses are available on 1 January, 1979. Brief descriptions and schedules follow.

The Microcourse

This course is designed for the technician, engineer, or manager who has had little or no experience with programmable microprocessor devices. The course will consist of lectures covering examples of architecture for a microprocessor, parallel I/O, serial I/O, interval timer, and interrupt controller. Laboratory sessions will be used to exercise the above programmable devices.

No prerequisite knowledge is assumed, but a knowledge of digital electronics and Boolean Algebra would be helpful.

STARPLEX™ User's Course

This course is designed for the individual who has recently obtained, or is considering the purchase of a STARPLEX™ Development System. The course will consist of lectures covering the features of the STARPLEX™ System and laboratory periods exercising the system and the points covered in the associated lectures.

Prerequisite knowledge includes either completion of the Microcourse or a knowledge of BASIC, FORTRAN, or 8080 Assembly Language.

Complex Peripherals

This course provides an introduction to the increasing family of microprocessor programmable peripheral devices for those individuals already involved in microprocessor system design. The course will consist of lectures on the capabilities, limitations, and interfacing of these peripherals. Software examples using the 8080 instruction set will be used to demonstrate the programming of these devices.

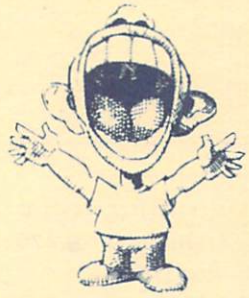
Familiarity with the 8080 instruction set would be helpful, but is not necessary. However, familiarity with some 8-bit bus-oriented microprocessor family is essential.

8060 SC/MP Applications

If you want to learn how to design a system using the popular SC/MP or INS8060 Microprocessor, or if you want to evaluate it for your application, this course will save you considerable time. Both hardware and software design techniques are discussed, including multiprocessing with the 8060. Laboratory sessions provide extensive hands-on time for program development.

COPS

This course is designed for the individual who has purchased or is evaluating the Computer Oriented Processor and development system for use in control applications. The course consists of lectures defining the COPS microprocessor and development system, and laboratories where applications examples are developed.



ENROLLMENT INFORMATION

Complete the enrollment form (p. 14) and mail it along with your tuition check. Please be sure to indicate the desired course, date, and location, as well as your name, company, address, and phone number. We will also send a map indicating directions to the training center. If a class is full you will be promptly notified and your tuition refunded (or applied to an alternate class if you desire). If you must cancel after enrollment please call the training center at least 5 days prior to the class and your tuition will be refunded in full.

Cancellations within 5 days of course start date are subject to a $50.00 charge.

If you have any questions, please call John Vorwerk at (408) 737-6453.

FIXING A BUG IN PACE BASIC

Bruce Edmundson
NSC, Santa Clara, CA.

This article explains how to fix a bug in the NATIONAL PACE BASIC interpreter (Rev. B). The following code results in a Range Error in line 30:

```
10 OPTION BASE 1
20 FOR I = 1 TO 10
30 T (I) = I
40 NEXT I
```

When a one-dimensional array is used with a base of 1, the second (internal) dimension of zero is not treated properly in calculations. The "size" is therefore computed to be zero, which results in the error message. Also, address calculations are wrong.

The problem can be fixed by merely adding the following statement to the program:

```
15 DIM T (10,1)
```

No other references to the array T need to be changed.

To patch the BASIC interpreter, insert new values in the following locations:

Address	Old Value	New Value
0006	————	F0E8
0007	————	C06F
0008	————	7801
0009	————	8000
00FA	————	F4E8
00FB	————	7901
00FC	————	98C9
0295	94C9	14FA
1103	E0F9	1406

ENROLLMENT FORM

NATIONAL SEMICONDUCTOR CORPORATION
Microprocessor Training, M/S 470
2900 Semiconductor Drive
Santa Clara, CA. 95051

Please enroll me:

Course Selected:

- ☐ The Microcourse
- ☐ STARPLEX User's Course
- ☐ Complex Peripherals
- ☐ 8060 SC/MP Applications
- ☐ COPS

Training Center Location:

- ☐ Santa Clara, Calif.
- ☐ Bedford, Mass.
- ☐ Other: _____

First choice date: _____ Second Choice Date: _____

Name/Title: _____

Company _____

Address _____

City/State/Zip _____

Telephone (___) _____

Billing Address if different from the above _____

P. O. # _____

Company/Attn: _____

Address _____

City/State/Zip _____

- ☐ Enclosed is my tuition check for $_____ made payable to "National Semiconductor Corp."
- ☐ Please bill me. My billing address is shown above.
- ☐ I can't attend now, but keep me in mind for future microprocessor courses.

SCHEDULE

COURSE NAME	WEST COAST Santa Clara, Calif. (408) 737-6453	EAST COAST Bedford, Mass. (617) 275-8530
MICROCOURSE 4 1/2 Days Tuition: $525.00	January 22-26 February 26 - March 2 March 26-30 April 23-27	March 12-16 April 9-13 May 14-18
STARPLEX™ USER'S COURSE 4 1/2 Days Tuition: $525.00	January 29 - February 2 March 5-9 April 2-6 April 30 - May 4	March 19-23 April 16-20 May 21-25
COMPLEX PERIPHERALS 3 Days Tuition: $425.00	February 6-8 March 13-15 May 22-24	February 19-21 April 24-26
8060 SC/MP APPLICATIONS 4 1/2 Days Tuition: $425.00	February 12-16 April 9-13	January 8-12 March 26-30 May 7-11
COPS 3 Days Tuition: $325.00	January 16-18 February 20-22 March 20-22 April 17-19 May 15-17	January 23-25 February 27-29 April 3-5 May 1-3

COMPUTE LIFETIME MEMBERSHIP FORM*

Lifetime membership fee is only $15.00 (DLR 15.00 for Australia). Make checks payable to COMPUTE and mail check and this card to the nearest location:

UNITED STATES

COMPUTE/208
National Semiconductor Corp.
2900 Semiconductor Drive
Santa Clara, Calif. 95051

EUROPE

COMPUTE
National Semiconductor GmbH
808 Fuerstenfeldbruck
Industriestrausse 10
West Germany

AUSTRALIA

COMPUTE
NS Electronics Pty. Ltd.
Cnr. Stud Rd. & Mtn. Highway
Bayswater, Victoria 3153

Please Print

Name_____ Title_____

Company_____

Address_____

Date_____ Signature_____

***Subject to periodic requalification**

BULK RATE
U. S. POSTAGE
PAID
PERMIT NO. 317
SUNNYVALE, CALIF.

USER'S LIBRARY ORDER FORM

Program Number	Program Name	Check for Program Listings	Source Paper Tapes: Number of Tapes	Cost Each	Total Cost
	PACE PROGRAMS				
SL0015A	PACRAM			$ 5.00	$
SL0018A	CALCULATOR		NA		
SL0022A	NUMPRG			$ 5.00	$
SL0025A	PALM			$ 5.00	$
SL0026A	TABTAP			$ 5.00	$
SL0029A	BINBCD		NA		
SL0032A	DIVIDE		NA		
SL0033A	DELSEM			$ 5.00	$
SL0035A	PRNTLM			$ 5.00	$
SL0036A	BASCII			$ 5.00	$
SL0037A	JITTER			$ 5.00	$
SL0048A	BIORHYTHM			$ 5.00	$
SL0054A	8080-X			$25.00	$
SL0059A	PACEFLPT			$15.00	$
SL0061	8048-X			$25.00	$
	SC/MP PROGRAMS				
SL0027B	SC/MP MATH			$ 5.00	$
SL0039A	TAPE I/O			$ 5.00	$
SL0041A	SCSQRT			$ 5.00	$
SL0043A	NIBL-N		*	$15.00	$
SL0043A	NIBL-P		*	$15.00	$
SL0049A	REFLEX			$ 5.00	$
SL0047A	PLOT			$ 5.00	$
	SC/MP NIBL PROGRAMS				
SL0051A	DISASSEMBLER			$ 5.00	$
SL0060	NIBASM		NA		
	SC/MP MACRO PKG				
SL0052A	MDSMAL				
SL0053A	MDSCMPL			$ 5.00	$
	PDP-15 PROGRAMS				
SL0045A	PACE-X			$ 5.00	$
	PDP-8 PROGRAMS				
SL0046A	SC/MP-X		NA		
	PDP-11 PROGRAMS				
SL0057	PACE-X			$15.00	$
SL0058	SC/MP-X			$15.00	$

***Price includes manual, program listing, and paper tape load module**

NOTES:

1. Please make sure the programs you select are for the microprocessor you have.
2. There is no charge for program listings, but the number of listings per order is limited to three.
3. NA indicates not available.
4. The prices quoted are in DLR for Australia and in dollars for the United States and Germany.

NAME _____

TITLE _____

COMPANY _____

ADDRESS _____

CITY _____

COUNTRY _____

Fill out the form completely, make your check payable to COMPUTE, and mail to:

UNITED STATES
COMPUTE/208
National Semiconductor Corp.
2900 Semiconductor Drive
Santa Clara, CA. 95051

JAPAN
COMPUTE
NS International Inc.
Miyake Bldg. 6F, 1-9 Yotsuya
Shinjuku-Ku, Tokyo 160, Japan

EUROPE
COMPUTE
National Semiconductor GmbH
808 Fuerstenfeldbruck
Industriestrasse 10, West Germany

HONG KONG
COMPUTE
NS Electronics HK Ltd.
4 Hing Yip St., 11th Floor
Kwun Tong, Kowloon, Hong Kong

AUSTRALIA
COMPUTE
NS Electronics Pty. Ltd.
Cnr. Stud Rd. & Mtn. Highway
Bayswater, Victoria 3153
Australia